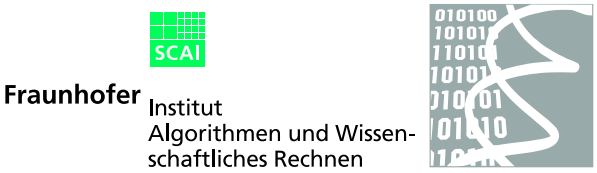
Pipeline Pilot Interface to *LeadIT*

Version 2.1.8.1

User Guide

(for *LeadIT* version 2.1.8 and above and Pipeline Pilot version 8.0 and above)

Edgar Derksen, Sally Hindle



The development of *LeadIT* began back in 1993 in the context of the project RELIWE funded by the German Federal Ministry of Education and Research (BMBF). In the following three years, Matthias Rarey and Stephan Wefing under the leadership of Thomas Lengauer laid the foundation of *LeadIT*. For his PhD thesis, Matthias developed the first prototype of *LeadIT*. Subsequently, as the head of the Computational Chemistry group at GMD, he guided the further development of *LeadIT* and its various extension modules. Today he is the Director of the Zentrum für Bioinformatik at the University of Hamburg and as such initiates a multitude of docking-related projects and provides guidance to the developers also in his role as a member of the board of BioSolveIT. Since 1996, many people have contributed to this code and the related research. This list is an attempt to acknowledge all co-workers in this ongoing project.
Developers (in alphabetical order):

- Holger Claussen (now BioSolveIT) developed FlexE to enable docking into ensembles of protein structures and provided a multitude of general enhancements in *LeadIT*
- Ingo Dramburg (now self-employed) developed a SMARTS™-based mechanism and chemical rules for systematic testing and manipulation of compounds
- Marcus Gastreich (now BioSolveIT) contributed significantly to an improved chemical model in *LeadIT*'s static data
- Sally Hindle (now BioSolveIT) developed FLEXSIS-PHARM, which allows docking to take place under constraints placed in the protein active site
- Andreas Kämper (now MPI) implemented various enhancements and extensions in *LeadIT*, e.g. the Tripos force field
- Bernd Kramer (now 4SC) developed parts of the scoring function and interaction model in *LeadIT*
- Markus Lilienthal (now BioSolveIT) implemented various post-optimization procedures and grid-enabling technology
- Gordon Müller (now BioSolveIT) is steadily improving the robustness of the software underneath the surface
- Matthias Rarey (now ZBH) developed the combinatorial extension module FLEXSIS^{C} adjunct for *LeadIT* (c.f. above)
- Frank Sonnenburg (now BioSolveIT) developed the Python-wrapper pyflexsis
- Stephan Wefing (now BioSolveIT) developed major parts of the physico-chemical models behind *LeadIT*

Other contributors (in alphabetical order):

Gerhard Barnickel (Merck KGaA)	Joachim Böhm (Roche)	Hans Briem (Schering)
Christian Buning (Sanofi-Aventis)	Gerhard Klebe (Univ. of Marburg)	Günther Metz (Santhera)
Thomas Mietzner (BASF)	Martin Stahl (Roche)	Gert Vriend (Univ. of Nijmegen)

Students involved in development and documentation (in alphabetical order) were: Christoph Bernd, Claus Hiller, Birte Seebeck, and Marc Zimmermann.

Institutions:
We (the developers) acknowledge GMD (now FhG) for supporting the work on *LeadIT* for nearly a decade and BMBF for funding *LeadIT*-related scientific work. We also thank our early cooperation partners BASF AG (Ludwigshafen), Merck KGaA (Darmstadt), and Boehringer Ingelheim (Biberach an der Riss) for enthusiasm, support and many fruitful discussions. Also, we are grateful for support from Tripos during many years of a fruitful developer-distributor relationship. Finally, we would like to thank all other companies and academic institutions who collaborated with us and helped us in a multitude of ways to create this great piece of software.

This document contains proprietary information of BioSolveIT GmbH and is protected by copyright. It is provided together with Software of BioSolveIT under a license agreement and may be used only in accordance with the terms and conditions of this agreement. The document serves solely for the purpose of using the Software. No part of the document may be transferred to any third party or reproduced as a whole or in parts without written permission from BioSolveIT.

Base software: © 2001 by Fraunhofer Gesellschaft (FhI-SCAI); Getline library: © 1993 by Chris Thewalt; PVM library: © 1997 by University of Tennessee, Knoxville TN; Python library: © 1991-1995 by Stichting Mathematisch Centrum, Amsterdam, The Netherlands; Torsion angle data: © by GMD SCAI, CCDC, BASF AG.

© 2015 BioSolveIT GmbH, An der Ziegelei 79, 53757 St. Augustin, Germany
Phone ++49-2241-2525-0, support@biosolveit.de

Contents

Contents **3**

1 Quick Start Steps **5**

1.1 Download and Import current *LeadIT* Package 5

1.2 Install *LeadIT* 5

1.3 Create or Update Protocols 5

2 Introduction **7**

2.1 About *LeadIT* 7

2.2 *LeadIT* Component in Pipeline Pilot 7

2.2.1 Binding mode prediction 8

2.2.2 Virtual high-throughput screening 8

3 Installation **11**

3.1 BioSolveIT Web Installation 11

3.2 Using the *LeadIT* Installation Component 11

3.3 Using Global Variables 12

3.4 Using Custom External *LeadIT* Installations 12

3.4.1 To Connect to an *LeadIT* Installation on the Pipeline Pilot Server . . . 13

3.4.2 To Connect to an *LeadIT* Installation on a Remote Linux Server 14

3.5 Running *LeadIT* in Parallel in Pipeline Pilot 15

3.5.1 Running in Parallel on the Pipeline Pilot Server 16

3.5.2 Running in Parallel on a Remote Linux Cluster 16

3.5.3 Example Scenarios and Required Settings 19

3.6 Uninstallation 20

4 Trouble Shooting **21**

4.1 Problems using the *ssh* Method 22

4.2 Further help and BioSolveIT PDF Reporter 22

5 Tips and Tricks **25**

5.1 Other Significant Parameters in the *LeadIT* Component 25

5.1.1 <Has Same File System> 25

5.2 Accessing Other Domains within Pipeline Pilot 25

1

Quick Start Steps

1.1 Download and Import current *LeadIT* Package

1. Download current "*LeadIT* in PipelinePilot" package on
   http://www.biosolveit.de/download/
2. Extract *LeadIT* package to a custom directory.
3. From the custom directory import all components (.xml) using PP-Client via drag and drop to your components collection.

1.2 Install *LeadIT*

1. Drag and drop *LeadIT* installation component from the components collection to an empty protocol.
2. Add your license for the executable of *LeadIT* in the field for the parameter <Run LeadIT -> on PP Server -> License Server or License File> in the Implementation tab – if your license is available from a license server, simply type in the name of the server in this format *@servername*, or if you have a license file then you may browse for it using the ... facility.
3. Run protocol once, it will result into a html report showing installation success.
4. For alternative installations and more details see chapter 3.

1.3 Create or Update Protocols

1. Drag and drop new components into a new protocol.
2. Update via a *right click drag and drop* of new component onto component reference in a protocol.
3. Now you are ready to setup the new components and run your protocol.

2

Introduction

2.1 About *LeadIT*

LeadIT is one of the most established protein-ligand docking tools in the literature. Cited hundreds of times, it has proved to be highly successful in numerous drug discovery applications – for references, please visit `http://www.biosolveit.de/references`. The main applications of *LeadIT* are binding mode prediction and virtual high-throughput screening (vHTS).

1. Binding mode prediction - For a protein with known three-dimensional structure and a small ligand molecule, *LeadIT* accurately predicts the geometry of the protein-ligand complex within a few seconds. To use *LeadIT* for binding mode prediction within Pipeline Pilot see the example in section 2.2.1.
2. Virtual high-throughput screening (vHTS) - *LeadIT* sets new records for vHTS. You can screen a library of approximately 1 000 000 compounds in about 8 hours on a 30-node cluster. For further references and information visit `http://www.biosolveit.de`. To use *LeadIT* for vHTS within Pipeline Pilot see the example in section 2.2.2 and setup for parallel processing in section 3.5.

2.2 *LeadIT* Component in Pipeline Pilot

You can see the general steps required in order to use the *LeadIT* Pipeline Pilot component in section 1; here we will cover the component parameter settings in more detail to help you fine tune your calculation. The component parameters and the error port are used as follows:

Project Filename Set the docking project for the calculation. The project is a file with `.fxx`-extension. You can find examples within this package or you can use the *LeadIT* GUI to create a new project.

Number of Solutions For each incoming molecule the *LeadIT* component runs a docking calculation using the defined project which results in a set of poses. These are ranked by docking score. Only the best <Number of Solutions> poses are output from the component.

Error port If the calculation for the input molecule fails, the input molecule will be passed to error port.

The following two examples form a first simple tutorial to see how the *LeadIT* component works.

2.2.1 Binding mode prediction

Figure 2.1: Starting example for binding mode prediction.

1. First you need a source for the input molecules. Drag and drop a molecule reader component like "Mol2 Reader" to your empty pipeline area. Select a MOL2 file in the parameter tab with the parameter <Source>. You can try out a MOL2 file in the examples folder of this package.
2. Set the <Maximum> parameter to 1 to ensure you start with only one molecule. This will form a quick example as a first test.
3. Drag and drop the *LeadIT* component to your pipeline area and connect the output port of the molecule reader to the input port of the *LeadIT* component. Go to the *LeadIT* component and define a project with the parameter <Project Filename> in the parameter tab.
4. By default 10 different docking poses will be calculated. You can change this value at the paramter <Number of Solutions> (be aware that *LeadIT* can create hundreds of poses).
5. To see a simple visual result of the poses with their scores, drag and drop the component "HTML Table Viewer" to your pipeline. Connect the output pass port of *LeadIT* component to the input port of "HTML Table Viewer".
6. Your workflow should look like the example in figure 2.1 and is ready to run.

2.2.2 Virtual high-throughput screening

1. Start with a molecule reader as described in 2.2.1. In this example the reader defines the library you want to screen. You can also replace the MOL2 reader with any molecule reader available in the Pipeline Pilot collection.
2. Remember to change the <Maximum> parameter to read all molecule records from the file.

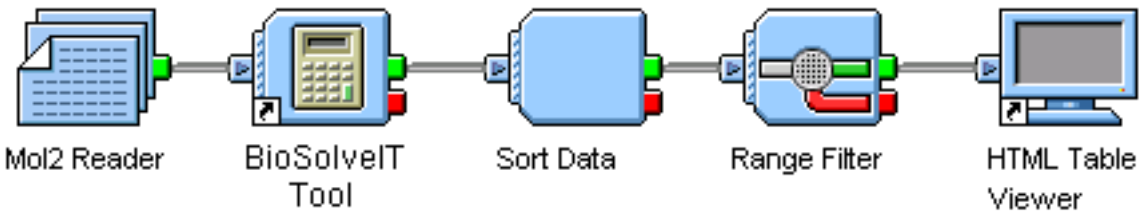


Figure 2.2: Starting example for virtual high throughput screening.

3. Set up *LeadIT* component as described in 2.2.1.
4. Change the <Number of Solutions> parameter to 1 (If you want to take more poses into account you have to do some extended analysis of the results *after* the docking calculation using the components available to you in Pipeline Pilot- check the Pipeline Pilotdocumentation for more information.) The result will now be one pose per input molecule.
5. The resulting poses need to be sorted to create a ranked list of your library. Drag and drop a "Sort Data" component and connect it to *LeadIT* component. Set the parameter <SortProperty1> to value "BSIT_*LeadIT*_Total_Score". The lowest scores are the best in *LeadIT* so set the parameter <SortDirection1> to "Increasing".
6. To select a subset of the resulting ranked list you can add a filter. In this example we want to filter the 100 best results. Drag and drop the component "Range filter" and connect it to the "Sort Data" component. Set the parameter <List> to "1-100".
7. Finally add a viewer or a writer component to view and/or save your results. Your pipeline may look like the example in figure 2.2. Change the workflow to your needs and, for example, try out some more detailed analysis.

3

Installation

3.1 BioSolveIT Web Installation

The easiest way to download and install BioSolveIT packages and tools is to download and run BioSolveIT Web Installer component. Download BioSolveIT in PipelinePilot package from http://www.biosolveit.de/download. Read UserGuide within that package for further details.

Note that both PP-Client and PP-Server need web access to run BioSolveIT Web Installer. Without web access you need to follow one of the following sections in this chapter.

3.2 Using the *LeadIT* Installation Component

The *LeadIT* components are set by default to use a so-called auto installation of *LeadIT*. This works as follows. When you run either of the *LeadIT* Calculator or Similarity components (and the parameter <Run LeadIT> is set to *on PP Server* and the parameter <Run LeadIT-> on PP Server -> Use> is set to *FTrees Auto Installation*), they search for a *LeadIT* installation in the directory:

```
<scitegic install directory>/public/bin/BioSolveIT/
```

and use this installation to run the calculation. A html report pupup shows if a *LeadIT* installation is not found in this directory.

Use *LeadIT* installer component to install *LeadIT* prior to run calculations. In case of a default setup you don't need administrator rights to do this as it is actually Pipeline Pilot that carries out the installation and not you as a user. In special cases your system administrator prevents installations by PP and you need to ask him to install *LeadIT* .

As this installation knows nothing about the licenses you may have for *LeadIT*, you have to supply the license information separately. This is supplied using the parameter <Run LeadIT -> on PP Server -> License Server or License File> as described above in the section 1.

The installation is carried out once only (for that *LeadIT* version) and only once per server not per user. The software is available to all users once it is installed.

3.3 Using Global Variables

Use global variables as an alternative to the *LeadIT* installation component decribed in section 3.2. Note that global variables will only be used if components are set to run auto installation (see. section 3.2) but no installation is done by an *LeadIT* installation component yet.

Goto Administratin Portal accessible through the Server Home Page at http://localhost:9944/ if PP-Server is installed on your local machine. The Server Home Page is also available through your PP-Client in menu <Help -> Server Home Page>. Use default name: scitegicadmin and password: scitegic if you have not changed it yet. Browse to <Setup -> Globals>. Add Custom Global Protocol Properties.

Note that defining parameters within your protocol will override global variables if they are named the same. Use parameters within your protocol instead of global variables only if you want to use a custom installation for that protocol.

The first variable file_server_*LeadIT*_executable: must be set to the full path of *LeadIT* executable available by the PP-Server. Example:

```
C:\Program Files (x86)\BioSolveIT\FTrees\ftrees.exe
```

The second variable file_server_*LeadIT*_config must be set to the full path of *LeadIT* configuration. Example:

```
C:\Program Files (x86)\BioSolveIT\FTrees\config_ft.dat
```

In comparison to *LeadIT* auto installation (see section 3.2) you need to check the compatibility between your defined installation and the components.

3.4 Using Custom External *LeadIT* Installations

You can also use an existing external installation of the *LeadIT* software. This means you also have the opportunity to use settings different to those set by default. To do this, you must already have *LeadIT* installed somewhere on your system outside of Pipeline Pilot. To install *LeadIT* yourself, visit the download page at BioSolveIT:
`http://www.biosolveit.de/download`
and fetch the download package for your system for the latest *LeadIT* package. Follow the instructions in the package to install *LeadIT* and receive your licenses. Enter the license information for *LeadIT* as described in the package **and not using the parameter <Run LeadIT -> on PP Server -> License Server or License File> as for the internal installation**.

To use an external installation of *LeadIT*, you must change the value of the parameter <Run LeadIT-> on PP Server -> Use> in the Implementation tab to *preinstalled LeadIT*.

There are actually two ways to use *LeadIT* with an external installation. These are by using *LeadIT* installed directly on the Pipeline Pilot server , or by accessing a remote machine where *LeadIT* is installed using *ssh* method. The method is selected using the parameter <Run LeadIT> Both methods are covered in more detail below.

3.4.1 To Connect to an *LeadIT* Installation on the Pipeline Pilot Server

The most common scenario is that you will have an installation of *LeadIT* on the Pipeline Pilot server. If you choose this option, you just have to enter the path to the executable and configuration file as parameters in the Implementation tab. Pipeline Pilot will then just start *LeadIT* whenever it is required by making a call to the executable that you entered.

This method is given the name *on PP Server* – On Pipelinepilot Server. You can see an example in figure 3.2.

- Requirement:
  - *LeadIT* is installed on the Pipeline Pilot server. You can see which machine is the Pipeline Pilot server by starting your copy of Pipeline Pilot Client on your own workstation and find the name or IP of the server shown at the bottom right of the status bar (see figure 3.1). You must find where the *LeadIT* installation is *on that machine*.

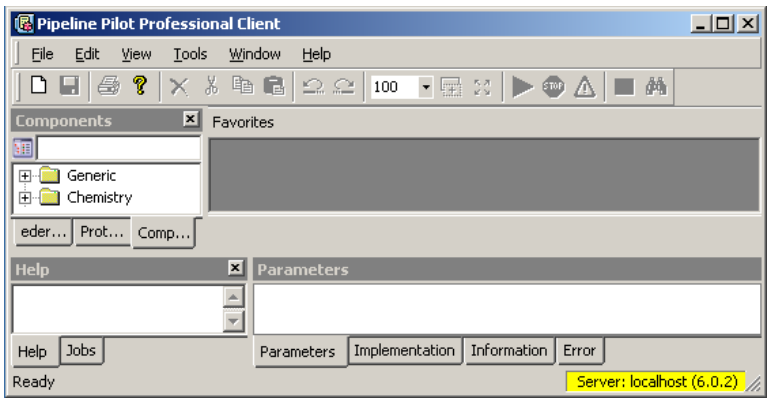


Figure 3.1: See where your Pipeline Pilot Server is installed.

- General Steps:
  1. Set <Run LeadIT> to *on PP Server*
  2. Set <Run LeadIT-> Use> to *Preinstalled LeadIT*
  3. For the parameter <Run LeadIT-> on PP Server -> Executable>, enter the path to the *LeadIT* installation on the Pipeline Pilot server. For example: `C:\Program Files\BioSolveIT\leadit3\leadit.exe`
  4. Add your license for the executable of *LeadIT* in the field for the parameter <Run LeadIT-> on PP Server -> License Server or File> in the Implementation tab – if your license is available from a license server, simply type in the name of the server in this format *@servername*, or if you have a license file then you may browse for it using the ... facility.

You can save these settings in the component.

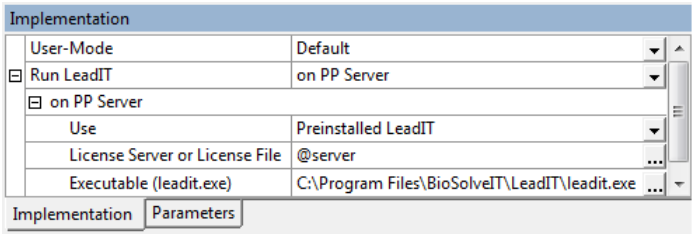


Figure 3.2: *PPServer*: Connect to an *LeadIT* Installation on the Pipeline Pilot Server

3.4.2 To Connect to an *LeadIT* Installation on a Remote Linux Server

Your existing *LeadIT* installation could be on a Linux computer remote from the Pipeline Pilot server – in this case we offer an alternative so you can use the remote installation instead. Here, the calculations carried out by *LeadIT* will be done on the remote Linux machine. Pipeline Pilot logs into the Linux machine using `ssh`, having copied all relevant files to the machine, and will run the calculation there, finally copying back all data it needs to the Pipeline Pilot server to continue with the pipeline.

This method is given the label *ssh*. You can see an example in figure 3.3.

- Requirement:
  - *LeadIT* is installed on a Linux machine available to the Pipeline Pilot server via `ssh`.
- General Steps:
  1. Set <Run LeadIT> to *via SSH*
  2. For the parameter <Run LeadIT-> via SSH -> Executable>, enter the path to the *LeadIT* installation on the Linux machine. For example: `/software/BioSolveIT/leadit/bin/leadit`
  3. For the parameter <sshHost>, enter the Linux machine host name
- User specific steps:
  1. For the parameter <Run LeadIT-> via SSH -> User>, enter the user login name for ssh on the Linux machine
  2. For the parameter <Run LeadIT-> via SSH -> Password>, enter the user password for ssh on the Linux machine
  3. There are more advanced options to be found under <Run LeadIT-> via SSH -> Options> for more specific ssh parameters. Note the option <Run LeadIT-> via SSH -> Options -> Delete Results>, this may be useful for trouble-shooting later.

You can save these settings in the component (be sure not to save your own user specific login details in components available to others!).

All files necessary for the *LeadIT* calculation will be transferred via `scp` between the Pipeline Pilot server and ssh Linux machine. Files copied and files created the remote server

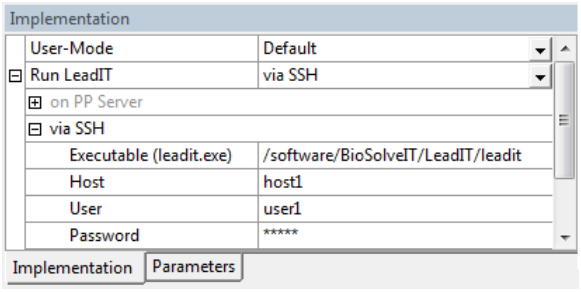


Figure 3.3: *ssh*: Connect to an *LeadIT* Installation on a Remote Linux Server via ssh

are automatically deleted at the end of the job leaving no trace. However, in case the user would like to leave a copy of the calculation and result files on the Linux machine, or for trouble-shooting as mentioned above, it is possible to set a parameter to tell Pipeline Pilot not to delete these files:
<Run LeadIT-> via SSH -> Options -> Delete Results> : *False*

3.5 Running *LeadIT* in Parallel in Pipeline Pilot

LeadIT in Pipeline Pilot takes advantage of the parallel computing options available in Pipeline Pilot to speed up longer calculations. This section tells you how to adjust the options to your system and needs. You will find the options in the Implementation tab, as in figure 3.4.

The most important limitation to a parallel processing calculation is the number of *LeadIT* licenses that you have. If you only have a single license then parallel calculations will not be possible. Further choices in the set-up of the parallel computing computing calculation depend on the number of Pipeline Pilot licenses you have plus your choice of connection to the external *LeadIT* installation (*on PP Server* or *via SSH*).

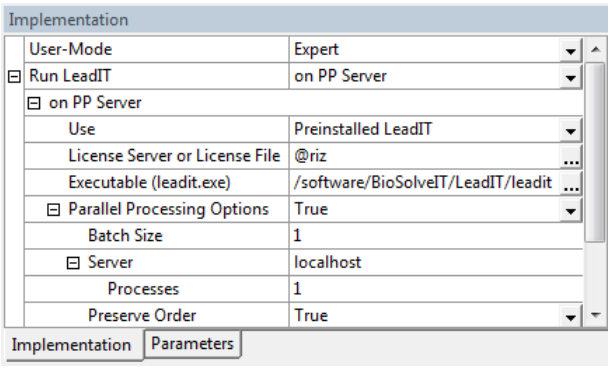


Figure 3.4: The options for tuning parallel processing are found in the Implementation tab

It is important to note that a balance must be achieved between the overhead caused by running several calculations instead of one and the size of the calculation – there is a lot

of overhead involved in sending all the essential data to different computers and collecting the results. For the *LeadIT* component, however, we still think parallel computing will work also for parallel jobs with batch size 1. Of course, these figures depend on the speed of your machines and network: experiment with your set-up if you intend to carry out large calculations often. Also we advise you to read the Pipeline Pilot documentation about parallel processing to understand more fully how it works.

Note: to set up large parallel processing jobs, you will need Administrator rights to change one setting.

3.5.1 Running in Parallel on the Pipeline Pilot Server

You may have a multi-processor machine as your Pipeline Pilot server. If you also have the appropriate number of Pipeline Pilot and *LeadIT* licenses the simplest way to start a parallel calculation is to raise the number of processes to the number of processors of the machine.

You could also have more than one Pipeline Pilot server available in your network. If so, you can enter a list of the server names at the parameter <Server>. Below that, for the parameter <Processes> enter a list of the number of processes each server should receive. The lists are both comma separated and must be in corresponding order. Remember to adjust also the <Batch Size> accordingly.

Note: the path to the external installation of *LeadIT* must be same on all servers!

3.5.2 Running in Parallel on a Remote Linux Cluster

We have developed an implementation in the component whereby a large cluster can be incorporated to run *LeadIT* jobs, without them having to be Pipeline Pilot servers. However, it must be a Linux cluster and the component must use the ssh method. You must also have enough *LeadIT* licenses available to the cluster.

The settings in the Implementation tab must be made as for running the ssh method, with two important changes. Instead of one <Run FTrees -> via SSH -> Host>, enter a comma separated list of the host names in the Linux cluster. Then, you must choose how many batchs in total you want the job to be split into and enter this total in the <Processes> parameter. Remember to change the <Batch Size> to fit with the number of processes.

Figure 3.5 may help clarify how the method works.

The Pipeline Pilot server will run with the number of processes given in <Processes>. However, in this case the processes the server receives are not the *LeadIT* calculations themselves but instructions for starting the ssh jobs. The server iterates through its 10 jobs each time spawning an ssh job on a Linux host. Beware that the Pipeline Pilot server does not know how many processors the Linux hosts have so make sure you choose the number of <Processes> to fit the number of Linux hosts and their number of processors respectively – be careful not to overload the Linux hosts.

Figure 3.6 shows the detailed settings required to set up the calculation as in figure 3.5.

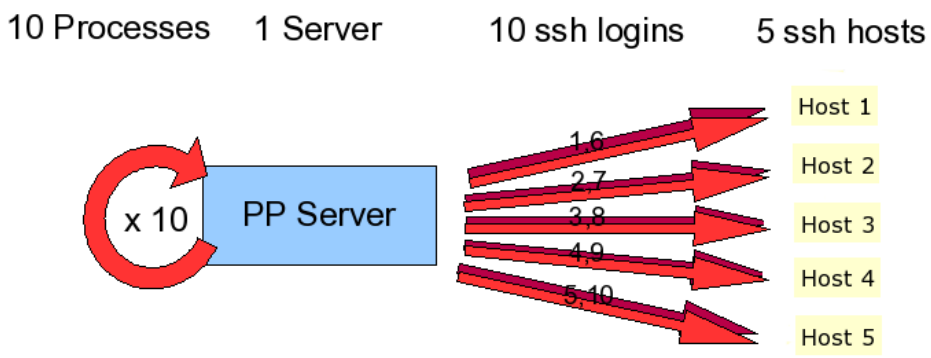


Figure 3.5: The method of parallel processing on Linux clusters in the *LeadIT* component. The Pipeline Pilot server is given 10 processes. The 10 processes start an ssh job respectively, distributed amongst the ssh hosts.

As you may already have realised, you could enter more than one Pipeline Pilot server at the <Server> parameter (along with another entry for number of <Processes> as a comma separated list) to execute a doubly parallel calculation!

Note: to get this method to work you will need to change the maximum number of processes per Pipeline Pilot server. The Pipeline Pilot Client will let you enter any number for the <Processes> parameter and does not warn you if this number is above the maximum.

Changing the maximum number requires Administrator rights! The number of processes per Pipeline Pilot server is usually restricted to the number of processors of the server. You must override this maximum to be able to set the number of processes you want for your parallel calculation. In the above example, the maximum must be set to 10 or more. Take the following steps:

- Go to the 'Scitegic Server Home Page', for example, via the Help menu in your Pipeline Pilot client.
- Click on 'Pipeline Pilot Administration Portal' and log in with the Administrator user name and password.
- In the last field of the table ('Maximum number of simultaneous parallel processing subprotocols allowed') change the value, click 'Save' and log out again.

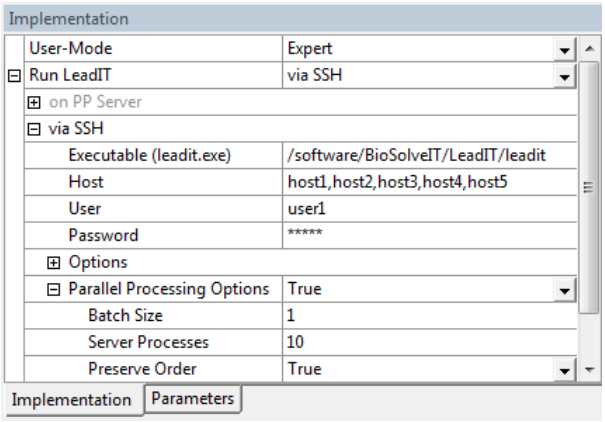


Figure 3.6: The options for parallel processing for the example shown in figure 3.5 would look similar to these

3.5.3 Example Scenarios and Required Settings

ssh/PPServer	*PPServer*
Parallel Processing Options	False
Number of PP servers in list	1
Number of ssh Hosts in list	-
Number of Processes	-
Behavior	The calculation will run as one complete job on the Pipeline Pilot server
ssh/PPServer	*ssh*
Parallel Processing Options	False
Number of PP servers in list	-
Number of ssh Hosts in list	1
Number of Processes	-
Behavior	The calculation will run as one complete job on the ssh host
ssh/PPServer	*PPServer*
Parallel Processing Options	True
Number of PP servers in list	1
Number of ssh Hosts in list	-
Number of Processes	1
Behavior	(Default) Induces the 'pipeline' effect: the job will run in chunks on the server processor(s)
ssh/PPServer	*PPServer*
Parallel Processing Options	True
Number of PP servers in list	1
Number of ssh Hosts in list	-
Number of Processes	>1
Behavior	A true parallel effect: the job will be run in chunks on the server processors
ssh/PPServer	*PPServer*
Parallel Processing Options	True
Number of PP servers in list	4 (each with two processors)
Number of ssh Hosts in list	-
Number of Processes	2,2,2,2
Behavior	A true parallel effect: the job will run in chunks in parallel split across 8 processors
ssh/PPServer	*ssh*
Parallel Processing Options	True
Number of PP servers in list	1
Number of ssh Hosts in list	5 (each with 2 processors)
Number of Processes	10
Behavior	A true parallel effect: the job will run in chunks in parallel split across 10 processors

3.6 Uninstallation

Use the *LeadIT* installer component to uninstall previous *LeadIT* versions by switching Parameter "Uninstall Previous Versions" to "True".

Use BioSolveIT Uninstaller component to uninstall all *LeadIT* versions.

Note that this kind of deinstallation only removes *LeadIT* versions installed by *LeadIT* installer component or BioSolveIT web installer component.

4

Trouble Shooting

The most commonly seen problems with *LeadIT* in Pipeline Pilot is with the connection to the external *LeadIT* installation. For one thing, *LeadIT* itself must be correctly installed on the system independently of Pipeline Pilot – it is essential first to make sure this is the case (especially to make sure that *LeadIT* can locate the licenses). Once *LeadIT* runs fine on your system, the remaining key task is to make sure the path to the executable is correct within the Pipeline Pilot component.

When the error messages pop up, they may contain an LeadITError in the error message box, as in figure 4.1.

Go to the 'Jobs' tab below the Protocol workspace and check under the last run job for a file called 'LeadITComponent Debug' – as in figure 4.2. Clicking on the link brings up HTML report with input and output data in a browser.

A correctly started *LeadIT* job outputs the following header: if there is a problem you will see some of this header and the point where the problem occurs:

```
                                L e a d I T
Copyright               Prediction of Protein-Ligand Interactions

BioSolveIT GmbH         Version:   1.0.0.1
An der Ziegelei 79      Modules:   [CDOCK] [FLEXE] [PHARM] [SCREEN] [PERMUTE]
53757 St. Augustin
Germany                 Original Author:    Matthias Rarey
www.biosolveit.de       Contact:            leadit@biosolveit.de

 For information about additional contributors and copyright notes
 please consult the user guide or type 'help about'.

 >> Running on DELTA (Windows 5.1) with 2 processors.
 >> Loaded settings (v 1.1.0) from 'C:\userdir\.leadit\settings.pxx'.
 >> LeadIT_base license check (BioSolveIT keys): succeeded.
 >> Licensed modules: LeadIT [CDOCK] [FlexE] [PHARM] [PERMUTE] [SCREEN] [DEVELOP]
...
```

4.1 Problems using the *ssh* Method

You may also experience problems using the *ssh* login, for example, the user name is unknown or the host is not found.

4.2 Further help and BioSolveIT PDF Reporter

More complicated errors may arise during the running of *LeadIT*. Again though, the errors will be collected and as much information shown as possible. If you are familiar with *LeadIT* you may want to take a look at all the output of the job yourself to see if you can recognize the problem. In this case, you can look in the temporary folders Pipeline Pilot sets up internally to find the output, or if you are working with the *ssh* method, set the parameter <Run FTrees -> via SSH -> Options -> Delete Results> to *False* so that you may then find the files retained on the *ssh* host: these will be in the directory set under the ssh parameter <Run FTrees -> via SSH -> Options -> Temp Path> (see the help text associated with this parameter to find its default value) – essentially a cryptically named folder whose name begins with the date and time of the job!

If you still do not know what is causing the errors, write down as much information as possible relating to your installation scheme. You can also create a PDF Report unsing BioSolveIT PDF Reporter component which summarizes installation data into one pdf file. Send all the information to:

support@biosolveit.de

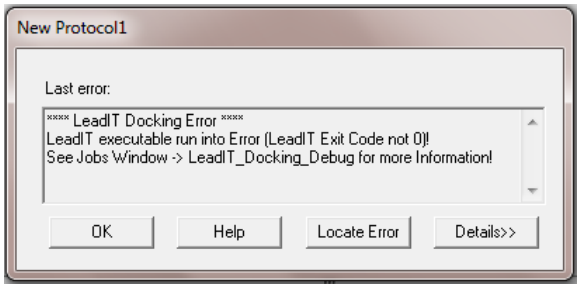


Figure 4.1: An error box reporting that the *LeadIT* exited with error.

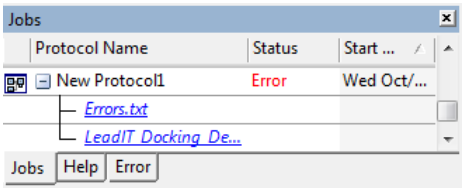


Figure 4.2: The full error report can be found in the 'Jobs' tab.

5

Tips and Tricks

5.1 Other Significant Parameters in the *LeadIT* Component

For detailed documentation of all parameters, refer to the documentation you find in the *Help* area of the Pipeline Pilot window.

We list here particularly interesting parameters: those that greatly influence the workflow or change the outcome of calculations, or those that may help you understand what is happening in the component.

5.1.1 <Has Same File System>

Implementation : <Run LeadIT-> via SSH -> Options -> Has Same File System>

Normally, for an *ssh* job, Pipeline Pilot must first copy all the data required by *LeadIT* to the <Run FTrees -> via SSH -> Host> using scp. This is time consuming. It is possible that the Pipeline Pilot Server and <Run FTrees -> via SSH -> Host> actually share the same file system rendering the scp process unnecessary. Select True if the Pipeline Pilot Server and <Run FTrees -> via SSH -> Host> share the same File System - no copying of data is necessary. Selecting False means Pipeline Pilot copies all the data to and back from the <Run FTrees -> via SSH -> Host> – this is just a little slower but will always still work. Leave the parameter set to False if you are uncertain!

5.2 Accessing Other Domains within Pipeline Pilot

Often in house data or even your own working data are accessible from a windows computer via a domain (a path starting for example 'z:\...' or '\\...') which you cannot find from within Pipeline Pilot. That means you must first literally transfer the data to the Pipeline Pilot Server itself.

If you are using a Linux Pipeline Pilot server, this hint does not apply.

To get around this problem and make the Pipeline Pilot working environment much more flexible you can allow users access to domains – you need Administrator rights to be able to do this! Also check first that you should change these settings as they may have already been set to fit the current environment.

- Go to the 'Scitegic Server Home Page', for example, via the Help menu in your Pipeline Pilot client.
- Click on 'Pipeline Pilot Administration Portal' and log in with the Administrator user name and password.
- In the Security tab go to Authentification.
- For the 'Authentification Method' choose 'DOMAIN' and a set of parameters will appear.
- Enter the domain name in the field 'Domain' and choose 'Full' for 'Impersonation'
- Choose 'DOMAIN' for 'Retrieve Groups' and leave 'Limit access to listed domains' set to 'No'

- click 'Save' and log out again.

After you have done this you will need to enter your domain login details when you start the Pipeline Pilot Client.